REMOTE ACCESS TO EARTH SCIENCE DATA BY CONTENT, SPACE, AND TIME

Elaine R. Dobinson and Robert G. Raskin*
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
Elaine.Dobinson@jpl.nasa.gov and Robert.Raskin@jpl.nasa.gov

Abstract

For the past several years a small team of developers at the Jet Propulsion Laboratory and at the University of Rhode Island have been working on a system to provide Earth scientists with access to remotely held data sets stored in various standard formats as though the data were local to the user's application. This capability requires not only interoperability among multiple formats and remote access to data via the internet, but also the ability to identify and select small subsets of data by space, time, and measured parameter. This demo presents the combination of an http-based client/server application that facilitates internet access to Earth science data coupled with a Java applet GUI that allows the user to graphically select data based on spatial and temporal coverage plots and scientific parameters. Access to data by values of the measured parameters is feasible via the same indexing schemes used for space and time, and work to include this capability is in progress.

1. Introduction

The job of Earth science research today entails a search for relevant data, the acquisition of the data to a local site, the reading and translation of the data to usable formats, and, finally, the analysis and display of the data of interest, which may be a very small fraction of all of the data acquired. With the advent of very large data volumes from satellite instruments, and multiple distributed data archives and research institutes holding data, this process becomes cumbersome and laborious.

The Earth Observing System Data and Information System (EOSDIS), under development by NASA to archive and distribute data from its EOS instruments, has attempted to provide a one-stop shopping tool for its archives, but can only provide fulfillment of orders for entire files of data, either electronically via ftp or on hard media. Remote access to subsets of files stored in several of the commonly used Earth science formats is still unsupported in EOSDIS. To help solve this problem developers at the University of Rhode Island and the Massachusetts Institute of Technology [1, 2], and at the Jet Propulsion Laboratory [3] have been building a Distributed Oceanographic Data System (DODS) and a companion Earth Science Remote Access Tool (ESRAT). DODS provides a common data format with translators for many standard formats (netCDF, HDF, MATLAB, DSP or JGOFS) and data structures (array, swath, grid, etc.). A DODS server translates from the data structures of these formats into the intermediate DODS model. A DODS client performs the reverse translation, such as into MATLAB or netCDF. As a result, local applications expecting local data in a particular format can access remote datasets in other formats.

Additions to the basic DODS system were built at the Jet Propulsion Laboratory. A DODS/HDF server was built as part of this effort and supports the SDS, raster, Vdata and Vgroup elements of the HDF data model. A Java GUI was added to enable graphical search and discovery of data using spatial and temporal constraints. To link the Java GUI to the DODS core system, catalogs containing coverage indexes for oceanographic datasets were developed which map spatial and temporal coordinates to DODS array locations. This permits a DODS client to query by geographical and temporal coordinates and receive back only the data requested.

2. Earth Science Remote Access Tool

The resulting system, the Earth Science Remote Access Tool, is an end-to-end client-server system that supports data discovery, access to data in multiple formats, server subsetting, http-based data transfer, and integration with standard visualization and analysis tools. The overall ESRAT system architecture has been described previously in [3]. The software architecture is shown here in Figure 1.
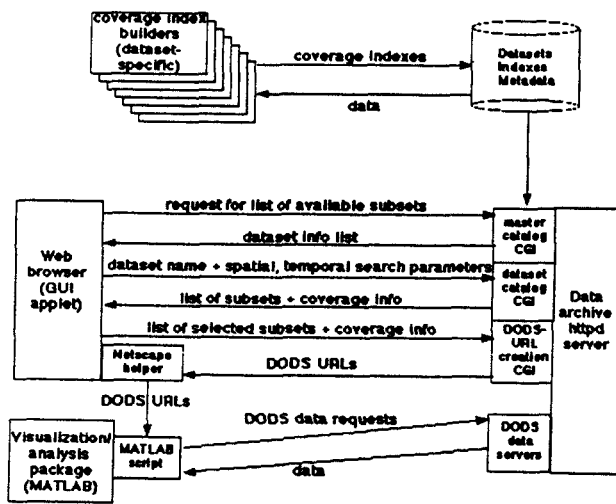

Figure 1. ESRAT Software Architecture

2.1 Data Discovery

ESRAT enables users to discover data of interest based upon specified spatial, temporal, and variable name constraints. ESRAT's data catalogs maintain spatial/temporal indices that allow users to extract only the desired portions of a file, without the need to know about file names, data structures, or array indices. A Java GUI applet is used for selection of spatial/temporal regions of interest and display of data coverages. The spatial/temporal coverages are displayed visually as layers that can be interactively selected. Java was used to increase portability and to enable access from any Web browser.

Although ESRAT is generic to most Earth science applications, the demonstration was tailored to a specific JPL application - the calibration/validation of the NSCAT instrument. The current data holdings include NSCAT wind measurements along with buoy and model data that can provide corroborating data for NSCAT. This enables the locations of NSCAT and ground truth

data sets (such as the TOGA/TAO buoys) to be readily overlaid, enabling scientists to visually discover overlapping, colocated data. AVHRR sea surface temperature data are also included in the data holdings. Figure 2 shows an example of a coverage plot depicting the presence of in-situ buoys, NSCAT orbital swaths, and AVHRR gridded sea surface temperature data over a selected spatial region and time interval.

2.2 Access to Data in Multiple Formats

ESRAT uses the Distributed Oceanographic Data System (DODS) to provide interoperability with data in multiple formats. DODS provides the capability to translate between the data structures of the supported format and an intermediate DODS data model. The data model for this format includes the constructor types: "Arrays", "Grids", "Sequences", and "Functions". A Grid is a georeferenced Array and need not refer to equally spaced points, as its name implies. A Sequence holds records in relational tables. A Function is a subclass of a Sequence where independent and dependent variables are explicit. The data model also supports "Structures" to create arbitrary combinations or hierarchies of the basic constructor types, analogous to a "struct" in the C programming language.
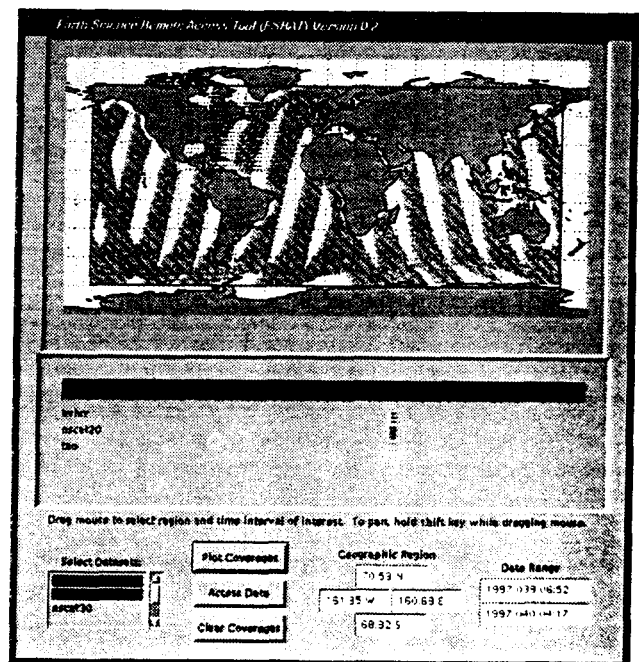

Figure 2. ESRAT Data Coverage Plot

Data structure information is stored in the Dataset Descriptor Structure (DDS), which is similar to a

structure declaration in C. The server uses the DDS to translate into the DODS data model. Some clients can use this information directly; for example, netCDF can use the DDS to identify variable names. All other metadata are packaged in a Dataset Attribute Structure (DAS). Both the DDS and DAS are ASCII files that are returned with any data request in the form of separate mime documents.

To add a format not presently supported, translation between the format's data model and the DODS data model must be provided. To add a new instance of a currently supported dataset, spatial/temporal mappings must be generated for the GUI; this is a relatively straightforward task. To add a new client type, the client's API must be expandable to enable linking with DODS remote access and translation software.

2.3 Server Subsetting.

ESRAT enables spatial/temporal subsetting on the data servers prior to data transmission, thereby reducing bandwidth requirements. The servers include the master catalog, dataset catalogs, and "URL creation" servers, implemented as C++ CGI programs. The master catalog contains a list of dataset holdings at local and remote sites and their associated URLs. The dataset catalogs contain the spatial/temporal bounds of each data subset and include examples of swath, grid, and point data. For swath data, each cross-track is individually cataloged in space and time. When a user requests swath data, the data server concatenates any contiguous cross-tracks satisfying the selection criteria, and returns a swath polygon to the client. The URL creation CGI assembles a constrained URL that is ultimately passed by the client to the visualization/analysis package via a Netscape helper program.

The client data request is sent to the server in the form of a constrained URL, corresponding to the user's spatial/temporal and variable name constraints selected in the GUI. The data server software parses the constrained URL, translates the formatted data, extracts the desired subset of interest (as specified in the constraint expression), and returns the data as a mime document (of type binary/x). A visualization/analysis package is linked with DODS client software that sends the data request and translates the incoming data into data structures understandable by the analysis package (e.g. MATLAB arrays).

Extending the dataset catalogs to include statistical summaries of the values of dependent variables is the next step to enable search by content. The current catalogs support searching by time, latitude, and longitude range. For content-based search, we could specify a range for the median (or other percentile) value of a dependent variable, such as wind speed. Thus, if a user is looking for all of the days where the winds were calm, the system would return those data subsets meeting that criterion.

2.4 Http-based Data Transfer

Immediate http-based data access to desired data subsets is provided using the DODS protocol. Unlike in the EOSDIS Core System, which provides only metadata about the files, ESRAT transfers the actual data subsets of interest directly to the user's application program, thus avoiding the need to download files.

2.5 Integration with standard visualization and analysis tools

ESRAT is intended to work with existing application programs. Data analysis and/or visualization is achieved by linking the client's application package (e.g. MATLAB) with DODS software. This software enables remote data access and provides translation from the intermediate DODS data structures to those supported by the client. This scheme allows data from multiple sites and in multiple formats to be loaded into a single DODS-compliant client (e.g. MATLAB), as shown in Figure 3. In the traditional configuration (a), the user's application program is linked to the standard API to read local data. In (b), the modified MATLAB/DODS API is used that enables either local or remote data access. No change in the application program is required other than relinking it with the modified API.
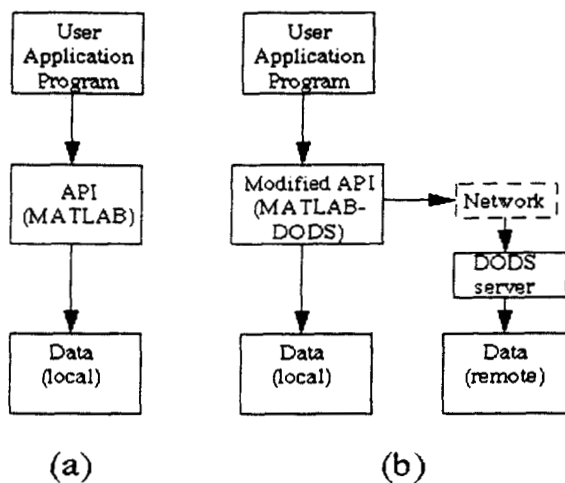

(a)   (b)

Figure 3.   Integration with MATLAB

3. Lessons Learned

ESRAT was developed as a prototype system, and as such, has provided many lessons learned:

1) Complete format interoperability is very difficult. 100% interoperability between all formats is very hard to achieve. However, partial interoperability is sufficient for most Earth science applications, as certain translation combinations are not likely in practice. HDF data structures are richer than those in netCDF, so some mappings from HDF into netCDF lose information. The DODS intermediate data model itself cannot handle common data types such as long integers, short floats, or unsigned variables, and this further hinders complete interoperability.

2) Space-time mappings are essential for Earth science. Catalog, directory, and search services were needed as additional components to DODS to construct the properly constrained DODS URL from the user's geophysical search parameters. Search and access by geophysical parameters is necessary for satellite data. Coincident data discovery via a GUI is desirable for instrument calibration/validation as well as other uses. We developed the needed software to allow users to access data subsets by spatial, temporal, and geophysical parameters.

3) ESRAT/DODS extensibility is relatively easy. Adding new data to ESRAT/DODS is relatively simple if the data reside in a currently supported format and type. Mappings of the array indices to space/time must be provided and the mappings stored in a binary file. If data in a new format are to be added, the server must be able to read the new format and translate its data model to that of DODS. This involves considerably more work, but needs to be done only once for a new format. Requests for interfaces to analysis packages other than MATLAB have been received. A client interface for the IDL analysis package is similar to the LoadDODS program we already have for Matlab and will be done this year.

4) Technology changes fast. Since the initial prototyping period Java has emerged as a language of choice for network applications. Redesigning and reimplementing DODS in Java would improve its reliability and portability. This work is currently under design.

4. Future Plans

The ESRAT work was initially sponsored by the ESDIS Prototyping Office. Future plans, funded by the NASA ESIP program, are to provide support for the HDF-EOS format, whose files contain the spatial and temporal bounds as embedded metadata using prescribed naming conventions. In addition, we are beginning to redesign and reimplement the system in Java. Support for indexing on dependent variables, to aid content-based searches, and a DODS interface to the IDL science analysis package are also planned.

A white paper [4] describes this project in greater detail. There is also a chapter in [5] which explains the use of ESRAT for interoperability with geographic information systems. Software support was provided at various times by Todd Karakashian, Isaac Henry, David Hecox, Robert Raskin, Robert Morris, and Jake Hamby. Elaine Dobinson, Deputy Task Manager for the Physical Oceanography Distributed Active Archive Center (PO.DAAC) at JPL, is the task leader.

References

[1] Gallagher, J. and G. Milkowski (1995) "Data Transport Within the Distributed Oceanographic Data System", *4th International World Wide Web Conference Proceedings*, 691-700.

[2] Sgouros, T. (1996) *DODS User Guide, Version 1.0.* http://dods.gso.uri.edu/DODS/user/guide-html/guide.html

[3] Dobinson, E. R. and R. G. Raskin (1997) Remote access tool for earth science data, Proceedings of the Ninth International Conference on Scientific and Statistical Database Management Systems, Olympia, WA, August 1997.

[4] Karakashian T., R. Raskin, R. Morris, and J. Hamby (1998) Earth Science Remote Access Tool White Paper, http://dods.jpl.nasa.gov/wp .

[5] Raskin, R. G. and E. R. Dobinson (to appear, 1998) Interoperability with the earth science remote access tool (ESRAT), in Interoperating Geographic Information Systems, Kluwer Publishing Co.